# LCA-2™

## Lower Case for the Apple-II*

For **Revision 7** and newer Apple-II and Apple-II Plus only
(CPU boards **without** RAM Configuration Blocks)

Get the most from your computer with

**ENHANCEWARE™**

since 1978 by

**Dan Paymar**
91 Pioneer Place
Durango, CO 81301

*Trademark of Apple Computer Inc.

# LCA-2™

## LOWER CASE ADAPTER FOR APPLE-II*

This is the LCA Model 2. It is for the Apple-II with Revision 7 and newer CPU board only. These boards **do not** have the RAM Configuration Blocks (the small plastic blocks next to the row letters D, E and F along the left side of the CPU board, usually labeled 16K 16K 16K and described on page 70 of the Apple-II Reference Manual).

**WARNING:** Do not attempt to install the LCA-2 on a computer that **has** the RAM Configuration Blocks. The LCA-2 is likely to be destroyed, and this is **not** covered by the warranty. The LCA-1 is designed specifically for the Revision 6 or older Apple-II.

### INSTALLATION INSTRUCTIONS:

1. Turn off power and remove the power cord. Turn the computer on its top or side, and remove the three phillips head screws along each side of the base plate. Then set the computer right side up with the front edge overhanging the edge of the table, and remove the four screws along the front edge. Lift the case only enough to slide it back a few inches. Do not remove the case completely. Be careful not to pull out the keyboard cable plug.

2. Remove the foam pad from the LCA, and check for bent pins. Save the pad for use in storing the old character generator chip.

3. Carefully remove the 2316 chip from location A5 on the CPU board (the 24-pin chip next to the keyboard connector) by gently prying each end upward with a small screwdriver. Be sure to pry between the chip and the socket, not under the socket. **CAUTION -Avoid causes of static electricity while installing the LCA.**

4. Plug the LCA-2 into socket A5 (where the 24-pin chip was removed), being sure it is fully seated with all pins in the socket (no bent pins and no pins overhanging at either end). Pin 1 (identified by a white dot) **must** be at the left next to the "4" printed on the front edge of the CPU board.

5. Press the old chip (that was removed at step 3) into the black foam pad to protect it from static electricity, and store it in a safe place.

6. The case may now be replaced. (First check that the keyboard cable plug is still fully seated.) You may want to replace only a couple of screws before testing the LCA.

7 To see all 256 character codes displayed by your computer, run the following Applesoft* program:

```
1  CALL -936  DIM H$(16)  H$="0123456789ABCDEF"
2  VTAB 4  HTAB 13  PRINT H$  PRINT
3  FOR Y = 0 TO 15  VTAB (Y+6)  HTAB 10
4  PRINT MID$(H$,Y+1,1);
5  FOR X = 0 TO 15  COLOR = X  PLOT X+12,2*Y+10
6  COLOR = Y  PLOT X+12,2*Y+11  NEXT X,Y  END
```

The leftmost column of the display is the most significant hex digit, and the top row gives the least significant hex digit of each character code. This program is based on a contribution by David M. Sanders of Cupertino, California. For an explanation of how this program works, see "PRINT THE UNPRINTABLES" by Bert Kersey in the September, 1980, issue of *Call - A.P.P.L.E.*

*Trademarks of Apple Computer Inc
LCA, LCA-1, LCA-1a, LCA-2, and Lower Case Adapter are trademarks of Dan Paymar

## OPERATION:

The Lower Case Adapter affects only the text mode display of the Apple, extending the character generator to include the set of 32 lower case characters. Specifically, the LCA causes any byte containing 111 in the top three bits to be displayed as a lower case letter or as one of the standard ASCII symbols "`","{","|","}",or"~" A small block "■" is displayed if the byte is $FF. There is no conflict with most standard Apple software because the 111 combination in the top three bits was not previously used. The top two bits are used by the Apple, however, to select reverse video or flashing, so lower case letters cannot be displayed flashing or in reverse video.

## USING LOWER CASE:

The Apple's keyboard generates only upper case ASCII codes, and its monitor (in ROM) intercepts any lower case input and changes it to upper case. Therefore, it is necessary to use special means to enter lower case letters and the other new symbols.

Most popular editors and word processors, such as Applewriter II*, include facilities for lower case entry and optional lower case display. Others require a patch to allow displaying lower case on the screen. Patches are included in the back of this manual for Applewriter* 1.0 and Apple Pascal 1.1. We do not have patches for other versions of these products or for any other products, nor do we have patches to enable using the shift key with these products.

Some software products make no provision for lower case, and some of these are "protected" so that it is even impossible to patch them. Check the manual that should have been provided with the specific

software product to determine whether it will support lower case display. If the manual is not clear, or if the product will not support lower case, then check with the supplier of the software. Unfortunately, there is no way that Enhanceware can be of help in such a case.

For users who have a disk system, we have DICE (Dan's I/O Control Enhancements) available on a DOS 3.3 diskette. DICE makes it easy to enter and use lower case with most BASIC programs (either Integer BASIC or Applesoft*). DICE bypasses part of the monitor code to allow easy entry of lower case letters and special symbols from Apple's keyboard. With the latest version, DICE 83, either the ESC key or the modified shift key may be used for case shifting. DICE also provides several other enhancements to the Apple's I/O system, including variable speed slow list, entry of all special symbols, and easy cursor positioning. To use DICE, make a copy of the diskette and load your programs onto the copy. Each time you boot the diskette, the active routines (which are completely in 6502 machine code) are automatically loaded into a file buffer so that they are out of the way of your programs. There is no need of a HIMEN or LOMEN to avoid overlaying DICE, and only 8 bytes of zero page are used (locations 9 and $19-$1F). DICE should be available at your dealer where it may be copied for support of the LCA, or it may be ordered direct from Enhanceware for $10 postpaid. Instructions for modifying the shift key are included. The DICE INFO programs on the diskette give complete instructions for using DICE. Telephone orders are accepted at (303)259-3598 for C.O.D. shipment or using Visa or MasterCard.

The very similar DICE 82 is fully documented in the April, 1982, issue of *Call-A.P.P.L.E.*, with corrections in the April, 1983, issue. The shift key modification is documented in the May, 1982, issue.

# DEMO PROGRAMS

```
]LIST
100  REM  APPLESOFT LCA DEMO
110  REM       11-25-79
120  CALL  - 936: VTAB 5
130  DIM A$(255)
180 UCF = 2: REM UPPER CASE FLAG:
     0=LOWER CASE, 2=UPPER CASE,
     1=ONE UPPER CASE INPUT
190 A$ = "THIS PROGRAM DEMONSTRAT
     ES THE USE OF": GOSUB 1000
200 A$ = " THELOWER CASE ADAPTER
     WITH ": GOSUB 1000
201 A$ = "APPLESOFT": GOSUB 1000
210  PRINT " BASIC.".A$ = "THE SU
     BROUTINE AT LINE 1000": GOSUB
     1000
220 A$ = " CONVERTS AN ENTIRE STR
     ING (EXCEPT FOR": GOSUB 1000
230 A$ = " THE FIRST      CHARACTE
     R) AND PRINTS IT AS LOWER CA
     SE.": GOSUB 1000
240  PRINT    PRINT .A$ = "LINES 5
     00-699 GET CHARACTERS DIRECT
     LY  ": GOSUB 1000
250 A$ = " FROM THE KEYBOARD, AND
      THE UPPER CASE  ": GOSUB 10
     00
260 A$ = " FLAG DETERMINES UPPER
     OR LOWER CASE.": GOSUB 1000
270  PRINT    PRINT    PRINT "ESC",
     :A$ = " SWITCHES TO LOWER CA
     SE, OR TO UPPER CASE FOR THE
      NEXT CHARACTER ONLY." GOSUB
     1000
280  PRINT    PRINT    PRINT "ESC E
     SC";:A$ = " SWITCHES TO ALL
     UPPER CASE.": GOSUB 1000
290  PRINT    PRINT :A$ = "TRY IT
     NOW!": GOSUB 1000: PRINT    PRINT
300 H$ =  CHR$ (8): REM CTRL-H
500 A$ = ""  FOR I = 1 TO 255
520  GET C$:C =  ASC (C$)  REM
      GET AN INPUT CHARACTER
530  IF C = 3 THEN  END
540  IF C > 31 THEN 620
550  REM  CONTROL CHARACTER
560  IF C = 8 THEN 600: REM BACKS
     PACE
570  IF C = 13 THEN 680: REM RETU
     RN
580  IF C <  > 27 THEN 520: REM
     ESC
590 UCF = UCF + 1: IF UCF > 2 THEN
     UCF = 0
595  GOTO 520
600  IF I < 2 THEN 520
605  PRINT " ";H$;H$;" ";H$;
610 I = I - 1: IF I > 1 THEN 615
612 A$ = ""  GOTO 520
615 A$ =  LEFT$ (A$,I - 1): GOTO
     520
620  IF C > 63 AND UCF = 0 THEN C
      = C + 32: REM IF A LETTER O
     R SPECIAL SYMBOL AND IN LOWE
     R CASE MODE,  THEN CHANGE CH
     AR TO LOWER CASE
640  PRINT  CHR$ (C);  REM ECHO
650  IF UCF = 1 THEN UCF = 0: REM
      IF SINGLE UPPER CASE SWITCH
      BACK TO LOWER CASE
660  LET A$ =  LEFT$ (A$,I) +  CHR$
     (C): REM APPEND TO A$
670  NEXT I
680  PRINT "    REM ERASE CURSOR
690  PRINT A$: REM OPTIONAL
695  PRINT   GOTO 500
1000  REM  SUBROUTINE TO PRINT A
      STRING, CHANGING LETTERS TO
      LOWER CASE.
1010  REM STARTS AT CHARACTER #2.
     CHANGE 2 IN NEXT LINE TO 1
     TO START AT CHAR #1.
1020  FOR I = 2 TO  LEN (A$)
1030 A =  ASC ( MID$ (A$,I,1))
1040  IF A < 65 OR A > 90 OR I =
     1 THEN 1050
1045 A = A + 32: REM CHANGELETTER
      TO LOWER CASE
1050  PRINT  CHR$ (A);
1060  NEXT I
1070  RETURN
```

These programs (Applesoft* version at left and Integer version below) show how it is possible to accept entry of lower case from the standard Apple keyboard without special software or hardware modifications. However, these programs are presented for pedagogic purposes only and are not meant to be a practical means of using lower case. It is recommended that the DICE support software be obtained for simplified use of lower case with BASIC programs.

```
>LIST
  100 REM  LOWER CASE INPUT DEMO
  110 REM  5-17-79
  120 CALL -936: VTAB 5
  130 LENS=100: REM  LENGTH OF A$
  140 REM  CREATE A$, AND SET LOCA TO
      ITS LOCATION
  150 DIM A$(LENS):LOCA=256* PEEK
      (205)+ PEEK (204)-LENS-10
  160 REM  CREATE C$, AND SET LOCC TO
      ITS LOCATION
  170 DIM C$(1):LOCC=256* PEEK (205
      )+ PEEK (204)-11
  180 UCF=0: REM UPPER CASE FLAG -- 0=
      LOWER CASE, 2=UPPER CASE, 1=ONE
      UPPER CASE CHAR
  190 A$="THIS PROGRAM DEMONSTRATES TH
      E USE OF": GOSUB 1000
  200 A$=" THE LOWER CASE ADAPTER WITH
       INTEGER": GOSUB 1000
  210 PRINT " BASIC. ";:A$="THE SUBROU
      TINE AT LINE 1000": GOSUB 1000

  220 A$=" CONVERTS AN ENTIRE STRING (
      EXCEPT FOR": GOSUB 1000
  230 A$=" THE FIRST CHARACTER) TO LOW
      ER CASE. ": GOSUB 1000
  240 PRINT :A$="LINES 500-680 GET CHA
      RACTERS DIRECTLY": GOSUB 1000

  250 A$=" FROM THE KEYBOARD, AND THE
      UPPER CASE": GOSUB 1000
  260 A$=" FLAG DETERMINES UPPER OR LO
      WER CASE. ": GOSUB 1000
  270 PRINT   PRINT "ESC";:A$=" SWITCH
      ES TO LOWER CASE, OR CAUSES    T
      HE NEXT CHARACTER ONLY TO BE UPP
      ER      CASE. ": GOSUB 1000
  280 PRINT   PRINT "ESC ESC";:A$
      =" SWITCHES TO ALL UPPER CASE. "
        GOSUB 1000
  290 PRINT :A$="TRY IT NOW!": GOSUB
      1000: PRINT
  300 POKE LOCC,136:H$=C$: REM CTRL H
  310 POKE LOCC,255:X$=C$: REM CURSOR
  500 FOR I=1 TO LENS
  510 PRINT X$;H$;  REM SHOW CURSOR
  520 C= PEEK (-16384): IF C<128 OR
      C>223 THEN 520: REM   GET INPUT
      CHARACTER
  530 POKE -16368,0: REM CLEAR STROBE
  540 IF C>159 THEN 620
  550 REM CONTROL CHARACTER
  560 IF C=136 THEN 600: REM BACKSPACE

  570 IF C=141 THEN 680: REM  RETURN
  580 IF C<>155 THEN 520: REM ESC (IGN
      ORE OTHER CONTROL CHARACTERS)
  590 UCF=UCF+1: IF UCF>2 THEN UCF=
      0: GOTO 520
  600 IF I<2 THEN 520: PRINT " ";
      H$;H$;: REM  BACKSPACE
  610 POKE LOCC,0:I=I-1:A$(I)=C$:
       GOTO 510
  620 IF C>191 AND UCF=0 THEN C=C+
      32: REM   IF A LETTER OR SPECIAL
       SYMBOL AND IN LOWER CASE MODE,
       THEN CHANGE CHAR TO LOWER CASE
  630 POKE LOCC,C: REM C$=CHR(C)
  640 PRINT C$;  REM ECHO CHAR
  650 IF UCF=1 THEN UCF=0: REM IF SING
      LE UPPER CASE SWITCH BACK TO LOW
      ER CASE
  660 A$(I)=C$: REM APPEND CHAR TO A$
  670 NEXT I
  680 PRINT " ": GOTO 500
 1000 REM  SUBROUTINE TO CONVERT AN EN
      TIRE STRING (STARTING AT CHAR #2
      ) TO LOWER CASE
 1010 REM CHANGE 'LOCA+1' IN NEXT LINE
       TO 'LOCA' TO CONVERT THE ENTIRE
       STRING TO LOWER CASE
 1020 FOR I=LOCA+1 TO LOCA+ LEN(A$
      )
 1030 A= PEEK (I): REM  GET ONE CHAR
 1040 IF A<193 OR A>218 THEN 1060

 1050 POKE I,A+32: REM  CHANGE LETTER
      TO LOWER CASE
 1060 NEXT I
 1070 PRINT A$: REM OPTIONAL
 1080 RETURN
```

**TROUBLE-SHOOTING THE LCA-2:**

Although it is unlikely that you will need this section, it is included to assist you in case any problem should arise. This section has been included because the nature of most problems is such that they can be remedied by the store or by the user, thus saving shipping costs, and getting your system on the air more quickly.

With the LCA-2, there are a few errors that account for most problems encountered. If your LCA-2 fails to operate properly, please check for the following common errors before returning it for warranty service:

1 Chip installed backwards. Pin 1 (marked by a white dot) must be toward the front of the computer (next to the "4" printed on the front edge of the CPU board). Failure of the LCA-2 due to incorrect installation IS NOT COVERED BY WARRANTY

2. Bent pin. Check for the possibility of a pin outside of the socket or underneath the chip.

3. Chip offset. Check that all pins are in the socket. It is all too easy to insert this chip incorrectly such that pins are hanging over one end of the socket on the CPU board.

4. Wrong type LCA. If your computer has the RAM configuration blocks (see page 70 of the Apple-II Reference Manual), then it has a Revision 6 or older CPU board. Return the LCA-2 to your dealer. If it is still in new condition, then you may exchange it for a model LCA-1 by paying the price difference. Note that the LCA-2 may be destroyed by installation in an older model computer, and this is **not** covered by the warranty (see WARNING preceding installation instructions).

If the above efforts fail, please return the unit for warranty service. Use the black conductive foam over the pins to avoid static electricity damage as well as to prevent bent pins. This is a must since THE WARRANTY DOES NOT COVER PHYSICAL DAMAGE! If conductive foam is not available, press a piece of aluminum foil over the pins, then pack with sufficient padding to prevent bent pins. Return the unit to your dealer, or pack it **in a box** for shipment and send it postpaid to Enhanceware. Please include a **complete, detailed** description of the problem, a copy of your sale receipt, and **a statement that your computer is a revision 7 or newer model Apple-II.**

You may also call us at (303) 259-3598 for assistance between 9:00 a.m. and 2:00 p.m. mountain time on business days.

# APPLEWRITER* AND PASCAL PATCHES:

Applewriter* 1.0 may be modified for lower case display with the Dan Paymar Lower Case Adapter as follows:

1. Boot a DOS master, use COPY to make a copy of Applewriter, and put the **copy** into drive 1. Never modify an original!

2. 
```
>BLOAD TEDITOR
>CALL -151
* AE4.AE7
```
[The prompt is "]" on an Apple-II Plus*]
The prompt symbol will now be "*"
The computer should display
0AE4— A4 24 91 28

   If it displays anything else then stop; you probably have a different version of Applewriter, and this patch won't work.

3. Enter the patch as follows:
```
*AE4:20 25 18 EA
*152A:D0 06 A9 00 85 0B F0 E3 C9 88 F0 09 20 01 15
*1539:9D 00 02 20 33 18 48 A4 24 A9 A0 91 28 68
*1547:C9 8D F0 11 48 20 F0 FD A9 C0 85 0B 68 EA
*1825:A4 24 8D 23 18 20 33 18 91 28 AD 23 18 60
*1833:C9 20 B0 04 49 C0 30 15 C9 40 90 0F C9 A0
*1841:90 0D 29 BF C9 A0 10 07 49 60 4C 50 18 49 80 60
```

4. Press CTRL-C and RETURN to re-enter BASIC.
```
>UNLOCK TEDITOR
>BSAVE TEDITOR,A$803,L$1070
>LOCK TEDITOR
```

We do not have a patch for Applewriter 1.1 or for use of the shift key with Applewriter.

*Trademarks of Apple Computer Inc.

Pascal 1.1 may be modified for lower case display by entering and running the following Pascal program. Be sure to prepare a copy of SYSTEM.APPLE **before** running this program, and insert the **copy** at the appropriate point. Never modify an original diskette!

```
PROGRAM MODBIOS2;
VAR F:FILE
B: PACKED ARRAY[0..511] OF 0..255;
BEGIN
  WRITELN ('PLACE A COPY OF SYSTEM.APPLE');
  WRITELN ('IN DRIVE 4, THEN PRESS RETURN.');
  READLN;
  RESET (F,'#4:SYSTEM.APPLE');
  IF BLOCKREAD(F,B,1,5)=1 THEN BEGIN
    B[171]:=234; (* DAAB: NOP NOP NOP NOP *)
    B[172]:=234;
    B[173]:=234;
    B[174]:=234;
    B[388]:=76; (* DB84: JMP DB9C *)
    B[389]:=156;
    B[390]:=219;
  END;
  IF BLOCKWRITE(F,B,1,5)=1 THEN BEGIN
    WRITELN;
    WRITELN ('THE PATCH IS COMPLETE.');
    WRITELN ('PUT IN A SYSTEM DISKETTE,');
    WRITELN ('AND PRESS RETURN.');
  END;
  READLN;
END.
```

Now boot the copy and test it. Use it as your master whenever you need a new SYSTEM. APPLE diskette.

Copyright © 1983 by Dan Paymar